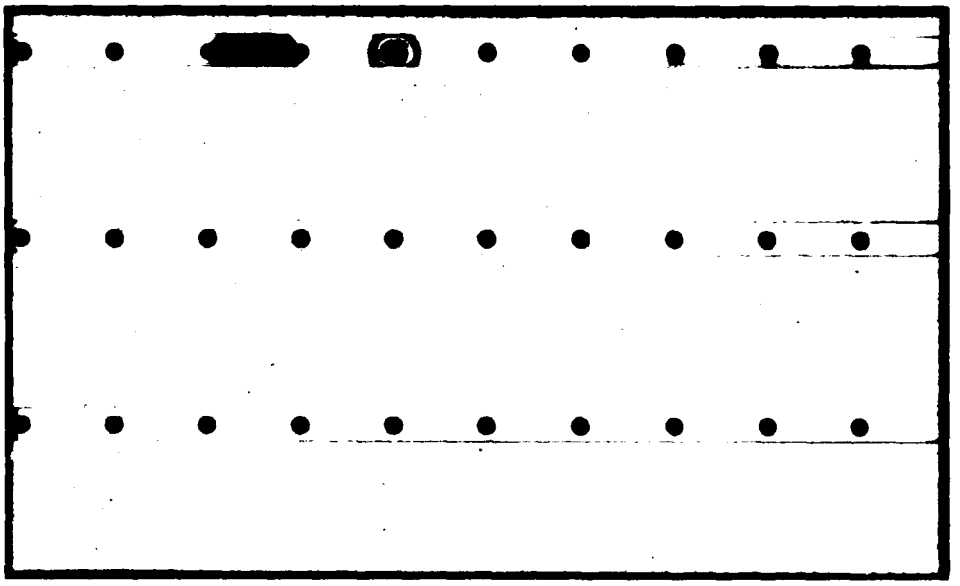
# Information and Computer Science



## TECHNICAL REPORT



UNIVERSITY OF CALIFORNIA
IRVINE

(4)

# Data-Driven Approaches to Empirical Discovery

**Pat Langley**
Irvine Computational Intelligence Project
Department of Information & Computer Science
University of California, Irvine, CA 92717

**Jan M. Zytkow**†
Computer Science Department
Wichita State University
Wichita, Kansas 67208

Technical Report 88-24

October 31, 1988



† Currently on leave of absence at Department of Computer Science, George Mason University, Fairfax, VA 22030.

This research was supported in part by Contract N00014-84-K-0345 from the Information Sciences Division, Office of Naval Research.

We would like to thank Herbert Simon, Gary Bradshaw, Bruce Koehn, and Bernd Nordhausen, who have contributed significantly to the development of BACON, FAHRENHEIT, and IDS.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

| REPORT DOCUMENTATION PAGE | | READ INSTRUCTIONS BEFORE COMPLETING FORM |
|---|---|---|
| 1. REPORT NUMBER<br>Technical Report No. 12 | 2. GOVT ACCESSION NO. | 3. RECIPIENT'S CATALOG NUMBER |
| 4. TITLE *(and Subtitle)*<br>Data-Driven Approaches to Empirical Discovery | | 5. TYPE OF REPORT & PERIOD COVERED<br>Interim Report 1/88–6/88 |
| | | 6. PERFORMING ORG. REPORT NUMBER<br>UCI-ICS-Technical Report-88-24 |
| 7. AUTHOR*(s)*<br>Pat Langley and Jan M. Zytkow | | 8. CONTRACT OR GRANT NUMBER*(s)*<br>N00014-84-K-0345 |
| 9. PERFORMING ORGANIZATION NAME AND ADDRESS<br>Department of Information & Computer Science<br>University of California, Irvine, CA 92717 | | 10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS |
| 11. CONTROLLING OFFICE NAME AND ADDRESS<br>Information Sciences Division<br>Office of Naval Research<br>Arlington, Virginia 22217 | | 12. REPORT DATE<br>October 31, 1988<br>13. NUMBER OF PAGES<br>28 |
| 14. MONITORING AGENCY NAME & ADDRESS *(if different from Controlling Office)* | | 15. SECURITY CLASS. *(of this report)*<br>Unclassified<br>15a. DECLASSIFICATION/DOWNGRADING SCHEDULE |

16. DISTRIBUTION STATEMENT *(of this Report)*

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT *(of the abstract entered in Block 20, if different from Report)*

18. SUPPLEMENTARY NOTES

To appear in a special issue of *Artificial Intelligence* on the topic of machine learning.

19. KEY WORDS *(Continue on reverse side if necessary and identify by block number)*

empirical discovery
data-driven heuristics
theoretical terms
intrinsic properties
history of science
numeric laws
scope of laws
qualitative processes

20. ABSTRACT *(Continue on reverse side if necessary and identify by block number)*

In this paper we track the development of research in empirical discovery. We focus on four machine discovery systems that share a number of features: the use of data-driven heuristics to constrain the search for numeric laws; a reliance on theoretical terms; and the recursive application of a few general discovery methods. We examine each system in light of the innovations it introduced over its predecessors, providing some insight into the conceptual progress that has occurred in machine discovery. Finally, we reexamine this research from the perspectives of the history and philosophy of science.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

# Introduction

In the last decade, a few researchers have turned their attention to a domain often considered the realm of genius – scientific discovery. The vast majority of this work has focused on empirical discovery, and much of the effort has been concerned with the discovery of numeric laws. In this paper we trace one evolutionary chain of research on discovery, in particular the development of data-driven methods relating to numeric discovery. We examine four systems – Gerwin's function induction system, Langley, Bradshaw, and Simon's BACON, Zytkow's FAHRENHEIT, and Nordhausen and Langley's IDS – and describe how each program introduces abilities lacking in earlier systems. The conceptual advances involve three different but interrelated aspects of discovery: the form of laws and theoretical terms discovered; the ability to determine the scope and context of laws; and the ability to design experiments. We evaluate each of the systems, but we focus on their theoretical contributions rather than on reporting their behavior in specific domains. We close the paper by reviewing the work on machine discovery from the views of the history and philosophy of science.

## Machine Learning and Discovery

One of the central insights of AI is that intelligence relies on large amounts of domain-specific knowledge. The field of machine learning is concerned with methods for acquiring such knowledge, and one approach to this problem involves machine discovery (Langley & Michalski, 1986). This approach can be distinguished from other work in machine learning by the degree of supervision provided to the learner. Some learning research focuses on direct instruction, in which a teacher gives explicit advice or declarative knowledge (e.g., Mostow, 1983). In work on learning from examples, the learner must acquire its own concepts or rules from experience, but the teacher preclassifies instances into useful classes (e.g., Dietterich & Michalski, 1983). Both of these approaches are supervised in that a teacher provides information that constrains the learning task. In contrast, discovery occurs in domains where no such teacher is available, forcing the learner to operate without supervision.

Within this view of discovery as unsupervised learning, one can further identify three different aspects of discovery which borrow from distinctions that occur within the philosophy of science. Some discoveries involve the organization of objects or events into categories and taxonomies; within machine learning, work on this problem generally goes by the name of conceptual clustering (e.g., Michalski & Stepp, 1983). Other discoveries involve the induction of descriptive regularities, some qualitative and others quantitative in nature. Finally, some discovery involves the formulation of explanatory theories. The dichotomy between description and explanation is actually a continuum, but one can identify extreme cases at both ends of the spectrum. For example, the ideal gas law has a clear descriptive flavor, relating the temperature, volume, and pressure of gas in a container. In contrast, the kinetic theory of gases, with its analogy to colliding balls, has a clear explanatory flavor.

Much of the research in machine discovery has focused on the induction of descriptive laws,[1] and in this paper we will limit our attention to that aspect of discovery. We can define the task of empirical discovery as:

- *Given.* A set of observations or data.
- *Find.* One or more general laws that summarize those data.

In the domains we will examine, an observation consists of a conjunction of attribute-value pairs, either numeric or symbolic in nature.[2] For instance, one might observe a particular combination of values for the temperature ($T$), volume ($V$), and pressure ($P$) for a contained gas. Laws take the form of relations (usually arithmetic) between these terms (such as $PV/T = k$ in the case of gases) and the conditions under which these relations hold. Empirical discovery is the task of finding such laws that account for a given set of data.

We have chosen to focus on empirical discovery in this paper for two main reasons. First, most research in machine discovery has dealt with this task, including our own work. Second, empirical discovery often occurs in the early stages of a field's evolution, before scientists have acquired much knowledge of the domain. As a result, it seems likely that general, domain-independent heuristics play a more central role in empirical discovery than in the process of theory formation and revision. Nonetheless, empirical discoveries are rare even among trained scientists, making them eminently worthy of attention. This combination makes them a good starting point for the mechanistic study of discovery.

## A Framework for Empirical Discovery

There are many paths to empirical discovery, but all of the systems we will describe in this paper share a common approach to this problem. Before describing the systems themselves, we should attempt to characterize this commonality. Taken together, the features that we will examine let one construct relatively simple and general discovery systems that still have considerable power.

First, all of the systems define *theoretical terms* that let them state laws in simple forms and that aid in the discovery process. The concept of momentum, defined as the product of mass and velocity, is one example of such a theoretical term. Using this product, one can state the law of conserved momentum as a simple linear relation. We will see that other types of theoretical terms are also possible. This can be viewed as a simple form of representation change, but we will not emphasize this aspect.

---

[1] There has also been considerable work on conceptual clustering (Michalski & Stepp, 1983; Fisher, 1987), and there have been recent efforts to use analogy in constructing explanatory theories (Falkenhainer, 1987; Langley & Jones, 1988).

[2] Some researchers (Lenat, 1977; Jones, 1986; Langley, Simon, Bradshaw, & Zytkow, 1987) have studied empirical discovery in domains involving more complex relational data, but we will discuss these only in passing.

Second, the systems all employ *data-driven* heuristics to direct their searches through the space of theoretical terms and numeric laws. These heuristics match against different possible regularities in the data and take different actions depending on which regularity they detect. Some heuristics propose laws or hypotheses, others define a new theoretical term, and yet others alter the proposed scope of a law. Different data lead to the application of alternative sequences of heuristics, and thus to different sets of empirical laws.

Finally, all the systems can apply their methods *recursively* to the results of previous applications, and they achieve much of their power in this fashion. Thus, knowledge resulting from the application of one heuristic can later be examined and extended by other heuristics. For instance, once a theoretical term has been defined, it can be used as the basis for defining still other terms. In general, this recursive structure leads to synergistic behaviors that would not otherwise occur.

## Alternative Frameworks for Empirical Discovery

For the sake of completeness, we should briefly consider some other frameworks for empirical discovery. Clearly, one might construct a discovery system that formulates laws without the aid of theoretical terms. For example, given one dependent term and a set of independent terms, one might use a regression algorithm to fit a curve to observed data. Such a law would directly predict the data with no intervening theoretical constructs. Few machine discovery systems operate in this manner; the construction of higher-level terms plays some role in nearly all AI work on empirical discovery. Moreover, all use some form of heuristic search in place of the algorithmic curve-fitting methods commonly used in statistics, and the notion of recursive application also plays a central role in most systems.

The mention of statistical methods raises an important issue. Statisticians have developed a variety of algorithms for summarizing data, which are firmly based in mathematics. Given the existence of these methods, why attempt to develop alternatives? One reason is that, historically, human scientists have not relied on such methods in making their discoveries. Even when AI does not attempt to model the details of human cognition, it generally borrows its inspiration from this area. Another reason is that AI methods, with their emphasis on heuristic methods, generally apply to broader range of tasks than algorithmic approaches. Thus, heuristic approaches to numeric discovery may handle a wider class of numeric laws than existing statistical techniques, and may even suggest methods for discovering qualitative laws.

Let us briefly review some other AI discovery work that has employed a heuristic search approach, but that differs from the methods we will describe in later sections. One example is Lenat's (1977) AM, which employed a variety of heuristics to direct its search in the domain of number theory. Starting with about 100 basic concepts such as sets, lists, equality, and so forth, AM used operators like specialization, generalization, and composition to generate new concepts. It then applied these operators to the resulting theoretical terms, eventually generating concepts such as multiplication, natural numbers, and prime numbers. The system also found qualitative laws that related these concepts, such as the unique factorization

theorem. Although AM clearly created new terms and applied its heuristics recursively, these heuristics were used primarily to *evaluate* new concepts rather than to create them. Thus, it should be viewed as a *model-driven* system rather than as a data-driven one.

Kokar's (1986) work on COPER shows that one can also apply the model-driven approach to numerical discovery. This system has considerable knowledge embedded into its generator, employing information about attributes' dimensions to generate a restricted class of theoretical terms. COPER then tests the resulting set of terms for consistency with the observed data. Its knowledge of physical dimensions lets it determine whether this set is complete and, if not, to search for additional (unobserved) terms. Once it has found a consistent set of higher-level terms, it searches a space of polynomial functions to find a numeric law that summarizes the observations. COPER has discovered the law of falling bodies and Bernoulli's law of fluid flow in this fashion.

Falkenhainer and Michalski's (1986) ABACUS takes a middle ground, basing its generation of new terms on using some knowledge of dimensions but also on a simple measure of correlation between variables. The latter heuristic has a data-driven flavor, making ABACUS more similar to the programs we will discuss later than either AM or COPER. The system also incorporates a component that clusters subsets of the data according to the laws they obey and uses these clusters to formulate conditions under which these laws hold. This component employs the Aq algorithm (Michalski & Larson, 1978) to search through the space of possible conditions; this process also has a mixed flavor, using some data to generate hypotheses and other data to evaluate them.

In the following pages, we describe four other discovery systems in greater detail. Given the variety of efforts on empirical discovery, our focus on a subset of this work deserves some justification. Our main reason is historical continuity. The four systems represent an evolutionary chain through the space of approaches to empirical discovery, with each system introducing innovations on its predecessors. We believe the evolutionary view reveals aspects of the discovery process that would remain hidden in a more traditional review. We also believe that the incremental development of AI systems, in which each program adds capabilities to previous ones, is an important methodological paradigm that deserves more widespread use. However, readers should not interpret our emphasis on these systems as downplaying the importance of other research in discovery.

## Gerwin's Model of Function Induction

Gerwin (1974) described one of the earliest machine discovery systems. He was concerned with inducing complex functions of one variable in the presence of noisy data. To this end, he collected and analyzed verbal protocols of humans solving a set of function induction tasks, as well as constructed a system that operated on the same class of problems. We will not review his experimental results here, except to note that he observed subjects using heuristic methods in their search for laws. The task itself and the system are more interesting for our purposes.

Gerwin's research on function induction introduced some important ideas that were to influence later work in empirical discovery. For instance, it served to clearly define the task of numeric discovery. At the same time, it also presented evidence that humans invoked heuristic search methods to solve such problems; the use of such methods (rather than algorithmic methods borrowed from statistics) made numeric discovery an interesting task for artificial intelligence.

Although Gerwin focused on functions of only one variable, some of his functions were quite complex. All were defined in terms of one or more primitive functions, taken from the set $e^{\frac{x}{2}}$, $x^2$, $x$, $x^{\frac{1}{2}}$, $\ln x$, $\sin x$, and $\cos x$, and combined using the connectives $+$, $-$, $/$, and $\times$. For instance, one such function is $y = x^2 \sin x + \ln x$; another is $y = x/\cos x$. However, a random component was included in each of the 15 test functions used, so the functions did not describe the data perfectly. In each case, Gerwin presented his subjects (and his program) with ten $x$ values and their associated $y$ values. From these data, the subjects and program were to infer the function best fitting the observations.

## Detecting Patterns and Computing Residuals

Gerwin's system included a number of condition-action rules for detecting regularities in the data. For instance, it looked for patterns having periodic trends with increasing (or decreasing) amplitudes; it also noted monotonic increasing (or decreasing) trends when they occurred. Each such pattern suggested an associated class of functions (or combination of functions) that could lead to its production. Thus, when the program noted a trend, it hypothesized that some member of the associated class was an additive component of the overall function.

Having identified a set of likely components, the system selected one of those functions and used it to generate predicted $y$ values for each $x$ value. It then subtracted the predicted data from the actual values, checking to see whether these *residuals* had less variance than the original observations. If not, the system tried some other function from the same class and repeated the process. If none of these were successful, it looked for some other pattern in the data.

Upon finding a useful component function, Gerwin's system applied the same induction method to the residual data. It looked for patterns in these data, proposed component functions, tested their effect, and either rejected them or included them as another component in the developing overall function. This process continued until the system could no longer detect any patterns in the residual data. Since no regularity remained, the program would halt at this point, assuming it had found the best description of the original data. Using this approach, Gerwin's program was able to discover many of the functions used in his experiment, some of them quite complex.

### Evaluating Gerwin's System

The particular system that Gerwin implemented relied on three important notions that we have already discussed. The first was the use of data-driven heuristics – his pattern-detecting condition-action rules – to direct the discovery process. The second was the notion of adding component functions, which can be viewed as a nascent form of theoretical term, and calculating residuals, which can be viewed as computing the values of those new terms. The final idea involved the recursive application of the original heuristics to these residuals, leading to new residuals and new data until a satisfactory function had been obtained. Taken together, these three features led to a simple yet powerful method for empirical discovery.

Despite its innovations, Gerwin's system was simplistic along a number of dimensions. It could discover only functions in one variable; it could define only one form of theoretical term; and its data-driven heuristics were specific to particular classes of functions. Moreover, the system had been tested only on a set of artificially generated functions, so its implications for real-world discovery tasks was not clear. Later work in machine discovery would address all of these issues.

## The BACON System

Although Gerwin's early work had many limitations, it provided an initial definition of the numeric discovery task and it suggested that this problem was amenable to the same heuristic search methods that had been used to explain other forms of intelligent behavior. These insights led directly to the BACON project (Langley, 1978, 1981; Bradshaw, Langley, & Simon, 1980; Langley, Bradshaw, & Simon, 1983), an attempt to construct a more general and more comprehensive model of empirical discovery.

### Representing Data and Laws

BACON is actually a sequence of discovery systems that were developed over a number of years. In this paper, we will focus on BACON.4, since that program incorporates the main ideas and tells the most coherent story. As input, the system accepts a set of independent and dependent terms. It can vary the values of the independent terms and request the corresponding values of the dependent terms. As an example BACON might be given three independent terms - the pressure $P$ on a gas, the temperature $T$ of a gas, and the quantity $N$ of the gas - and the single dependent term $V$, the resulting volume of the gas. Independent terms may take on either numeric or nominal (symbolic) values, whereas dependent terms are always numeric.

As output, BACON.4 generates three interrelated structures that constitute its empirical discoveries:

(1) a set of numeric laws stated as simple constancies or linear relations, such as $X = 8.32$ and $U = 1.57V + 4.6$, along with some simple conditions under which each law holds;

(2) a set of definitions that relate theoretical terms to directly observable variables, such as $X = Y/T$ and $Y = PV$; it is these definitions that let BACON state its laws in such a simple form;

(3) a set of intrinsic properties, such as *mass* and *specific heat*, that take on numeric values; these values are associated with the symbolic values of nominal terms; thus, the *mass* of *object A* may be 1.43 while the *mass* of *object B* is 2.61.

Although each structure has a very simple form, taken together they provide BACON with considerable representational power. Using these three knowledge types, the system has rediscovered a wide range of laws from the history of physics and chemistry, including forms of the ideal gas law, Coulomb's law, Snell's law of refraction, Black's law of specific heat, Gay-Lussac's law of combining volumes, and Canizzaro's determination of relative atomic weights. Now let us examine the process by which BACON accomplishes these discoveries.

## Discovering Simple Laws

BACON's most basic operation involves discovering a functional relation between two numeric terms. This is the direct analog to Gerwin's function induction task. For example, Galileo's law of falling bodies relates the distance $D$ an object is dropped from, to the time $T$ it takes to reach the ground. This law can be stated as $D/T^2 = k$, where $k$ is a constant. To discover laws relating two numeric variables, BACON employs three simple heuristics:

**INCREASING**
IF THE VALUES OF $X$ INCREASE AS THE VALUES OF $Y$ INCREASE,
THEN DEFINE THE RATIO $X/Y$ AND EXAMINE ITS VALUES.

**DECREASING**
IF THE VALUES OF $X$ INCREASE AS THE VALUES OF $Y$ DECREASE,
THEN DEFINE THE PRODUCT $XY$ AND EXAMINE ITS VALUES.

**CONSTANT**
IF THE VALUES OF $X$ ARE NEARLY CONSTANT FOR A NUMBER OF VALUES,
THEN HYPOTHESIZE THAT $X$ ALWAYS HAS THIS VALUE.

Table 1 presents some idealized data that obey the law of falling bodies. Given the cooccurring values of $D$ and $T$ shown in the table, BACON notices that one term increases as the other increases. This leads the INCREASING rule to apply, defining the ratio $D/T$ and computing its values. Since the resulting values increase as those of $D$ decrease, they lead the system to apply the DECREASING heuristic, which defines the product $D^2/T$. When it computes the values for this new term, BACON notes that all the values are very near the mean of 9.795. This causes the rule CONSTANT to apply, hypothesizing that $D^2/T$ always has this value; the system has rediscovered a form of Galileo's law. From this example, one can see that BACON makes no distinction between directly observable terms and those it has defined itself. The system can also discover other complex relations in this way, such

as Kepler's third law of planetary motion: $d^3/p^2$, where $d$ is the planet's distance from the Sun, $p$ its period, and $k$ is a constant.

**TABLE 1**
**Data obeying the law of uniform acceleration**

| *Time*($T$) | *Distance*($D$) | $D/T$ | $D/T^2$ |
|---|---|---|---|
| 0.1 | 0.098 | 0.98 | 9.80 |
| 0.2 | 0.390 | 1.95 | 9.75 |
| 0.3 | 0.880 | 2.93 | 9.78 |
| 0.4 | 1.572 | 3.93 | 9.83 |
| 0.5 | 2.450 | 4.90 | 9.80 |
| 0.6 | 3.534 | 5.89 | 9.82 |

## Discovering Complex Laws

In order to see how BACON discovers more complex laws involving a number of independent terms, let us consider a simple form of Black's heat law. This relates the initial temperatures of two substances ($T_1$ and $T_2$) with their temperature after they have been combined ($T_f$). The law can be stated as: $(c_1M_1 + c_2M_2)T_f = c_1M_1T_1 + c_1M_2T_2$, where $M_1$ and $M_2$ are the two initial masses and $c_1$ and $c_2$ are constants associated with the particular substances used in the experiment. For now we will assume the same substance is used in both cases; this makes $c_1 = c_2$ and lets us cancel them out from the equation. This gives a simpler form of the law: $T_f = (M_1/(M_1 + M_2))T_1 + (M_2/(M_1 + M_2))T_2$.

Given a set of independent terms such as $M_1$, $M_2$, $T_1$, and $T_2$, BACON constructs a simple factorial design experiment involving all combinations of independent values, and proceeds to collect data. In this case, the system begins by holding $M_1$, $M_2$, and $T_1$ constant and varying the values of $T_2$, examining the effect on the final temperature $T_f$ in each situation. In this way, the program collects the cooccurring independent and dependent values it requires to discover a simple law. In this case it finds the linear relation $T_f = aT_2 + b$, where $a$ is the slope of the line and $b$ its intercept. However, the system follows a conservative strategy upon discovering such a law, stating only that it holds when the other independent terms ($M_1$, $M_2$, and $T_1$) take on their observed values.

Nevertheless, BACON's ultimate goal is to discover a more general relation that incorporates all the independent variables. Thus, the system runs the same experiment again, but this time with different values for $T_1$, the temperature of the other substance. The result is a number of specific laws that hold for different values of $T_1$, but which all have the form $T_f = aT_2 + b$. At this point, the program shifts perspectives and begins to treat $a$ and $b$ as higher-level dependent terms, the values of which it has determined from the earlier experiments.

BACON then uses its methods for finding simple laws to uncover a relation between the values of $T_1$ and these two terms. In this example, the system finds that the slope $a$ is unaffected by $T_1$, which it states as the second level law $a = c$. It also discovers a linear relation between the temperature and the intercept that can be stated as $b = dT_1$; since the intercept of this line is zero, it is omitted.

Having established two second level laws, BACON now proceeds to vary $M_2$, the mass of the second substance, and to observe its effects on the parameters in these laws. This involves running additional experiments by varying $T_1$ and $T_2$, but once this has been done the system has a set of values for the parameters $c$ and $d$, each pair associated with a different value of $M_2$. Upon examining these values, BACON does not find any simple law but it notes that $c$ and $M_2$ increase together; as a result, it defines the ratio term $M_2/c$. This new term is linearly related to $M_2$, giving the third level law $M_2 = e(M_2/c) + f$. Similar regularities lead the program to define the product $dM_2$ and to find the linear relation $dM_2 = gd + h$.

**TABLE 2**

**Relations discovered at different levels for Black's law**

| Level | Term varied | Laws found | Laws implied |
|---|---|---|---|
| 1 | $T_2$ | $T_f = aT_2 + b$ | $T_f = aT_2 + b$ |
| 2 | $T_1$ | $a = c$<br>$b = dT_1$ | $T_f = cT_2 + dT_1$ |
| 3 | $M_2$ | $M_2 = e(M_2/c) + f$<br>$dM_2 = gd + h$ | $T_f = eM_2T_2/(M_2 - f)$<br>$+ hT_1/(M_2 - g)$ |
| 4 | $M_1$ | $f = jM_1;\ g = kM_1$<br>$h = lM_1;\ e = 1.0$ | $T_f = M_2T_2/(M_2 - jM_1)$<br>$+ lM_1T_1/(M_2 - kM_1)$ |
| 5 | $\text{Substance}_2$ | $j = pc_2;\ k = qc_2$<br>$l = rc_2$ | $T_f = M_2T_2/(M_2 - pc_2M_1)$<br>$+ rc_2M_1T_1/(M_2 - gc_2M_1)$ |
| 6 | $\text{Substance}_1$ | $pc_1 = -1.0$<br>$qc_1 = -1.0;\ rc_1 = 1.0$ | $T_f = M_2T_2/(M_2 + (c_2/c_1)M_1)$<br>$+ (c_2/c_1)M_1T_1/(M_2 + (c_2/c_1)M_1)$ |

Now that it has incorporated the independent terms $T_2$, $T_1$, and $M_2$ into its laws, BACON turns to the final variable, $M_1$. Varying this leads to a set of additional experiments in which the other terms are varied, and from these the system estimates values for the parameters $e$, $f$, $g$, and $h$ for each value of $M_1$. The slope term $e$ has the constant value 1.0 in all cases, but the remaining terms vary. Closer inspection reveals that $f$, $g$, and $h$ are all linearly related to $M_1$ and that each line has a zero intercept, with slopes respectively $j$, $k$, and $l$.

At this point, BACON has rediscovered the simplified version of Black's law presented above, though not in the form we specified. Table 2 traces the steps followed by the system, listing the laws formulated at each level of the discovery process; at this point of the discussion

we are at level 4. We should note that, as it finds laws at each level, the program places conditions on these laws corresponding to the values of those terms that it has not yet varied. As it incorporates these terms into higher-level laws, the conditions are generalized. Thus, BACON gradually expands the scope of its laws as it moves to higher levels of description. We will return to the issue of scope later in the paper.

## Postulating Intrinsic Properties

The above methods suffice to discover laws that relate numeric terms, such as occur in the ideal gas law. However, there are many historical cases in which scientists were also confronted with nominal or symbolic attributes. For instance, the two substances in Black's law are best described in this manner; one can combine water with water, water with mercury, and so forth. Upon varying the substances in this manner, one finds that the values of parameters in the various laws also change. However, one cannot incorporate such symbolic terms directly into its numeric laws; some other step is required.

BACON's response in such cases is to *postulate* numeric terms that are associated with the observable nominal ones; we call these *intrinsic properties.* In the Black's law example, one can introduce such a property (called *specific heat*), the values of which are associated with different substances. Thus, if we let the specific heat $c$ for *water* be 1.0, then the specific heat for *mercury* is 0.0332 and the specific heat for *ethyl alcohol* is 0.456. Once BACON has established these values, it can relate the values of $c$ to parameters from its various laws, giving a higher-level law that effectively incorporates the two substances.

Let us continue with the Black's law example where we left off. BACON had incorporated the numeric terms $M_1$, $M_2$, $T_1$, and $T_2$ into a coherent set of laws, all ultimately related to the final temperature $T_f$. The system had also arrived at values for four parameters at the fourth level of description. One of these (call it $i$) involved a simple constancy; the others, $j$, $k$, and $l$, were the slopes of linear relations. The values for these parameters were conditional on the particular pair of substances used in the experiment, in this case two containers of water.

BACON's next step is to vary the second substance, using different materials such as mercury and ethyl alcohol with the first substance (still held constant as water). Upon doing this, the system notes that the values of $j$, $k$, and $l$ all vary, though the value of $i$ remains unchanged. In order to incorporate these terms into a higher-level law, the program requires some numeric independent variable associated with the second substance; we will call this $c_2$. BACON must assign values for this term, one for each nominal value of the substance, and it bases these values on those for the parameter $j$ (though $k$ or $l$ would have served equally well). The term $c_2$ is an intrinsic property, and the numeric values assigned to it are intrinsic values. These are initially stored with the condition that the first substance be water.

At this point BACON notes a linear relation between $c_2$ and $j$, but this is tautological, since it had defined the intrinsic property using the values of the latter term. However,

the system also discovers linear relations between $c_2$ and $k$ and between $c_2$ and $l$; these are not guaranteed to hold and so have empirical content. The program has moved beyond tautologies and into laws capable of making predictions. Even more interesting events occur when the program varies the first substance in the experiment.

Upon placing mercury in contact with water, with mercury, and with ethyl alcohol, BACON finds that the values of the slope $j$ differ from when the first substance was water. But more important, they are linearly related to the earlier values of $j$. This tells BACON that the values of its intrinsic property should be useful regardless of the first substance; the condition that the first substance be water is dropped and the intrinsic values are stored with only the values of the second substance as a condition for retrieval. Thus, one value of $c_2$ is associated with water, another with mercury, and a third with ethyl alcohol. This lets the system retrieve the values of $c_2$ that it identified earlier and to note a linear relation between $c_2$ and the parameter $j$. Moreover, this law is non-tautological; the values of $c_2$ were based on earlier values of $j$, not the current ones.

This generalization of the conditions on the intrinsic values also proves useful at the next (and highest) level of description. Different linear relations occur when different substances are placed in the first container, and the slopes of these lines provide the dependent terms for BACON to relate to the first substance. Since this is a nominal term, one could define a new intrinsic property, but there is no need; the conditions on the property $c$ have been sufficiently generalized to let its values be used in this case as well. Thus, BACON infers the values of $c_1$ and relates these to the various slope terms. The final set of relations correspond to Black's heat law, and the terms $c_1$ and $c_2$ correspond to the specific heats of the first and second substance, respectively. Table 2 summarizes the forms of the final laws.

## Evaluating BACON

Now that we have examined BACON's representation and heuristics, we can evaluate its behavior in terms of some general issues relating to empirical discovery. Basically, we will conclude that on two dimensions – the forms of laws it can handle and the types of new terms it can define – the system performs quite well. However, the program's ability to determine the scope of laws and its ability to design leave much to be desired.

Recall that BACON states all laws as either simple constancies or linear relations between two variables. However, when combined with the ability to define new ratio/product terms and to introduce intrinsic properties, this is sufficient to state a wide range of laws. For instance, the system can formulate laws involving exponents; one example is Kepler's law ($D^3/P^2 = k$) and another is Coulomb's law ($FD^2/q_1q_2 = k$). Another is Ohm's law for electric circuits, which in its most general form can be stated as $TD^2/(LI - rI) = b$. BACON can also discover a general version of the ideal gas law that does not rely on the absolute temperature scale: $PV = aNT + bN$. These suggest that the system can discover a respectable variety of empirical laws.

BACON also fares well in its ability to define new terms, and as we have stated, much of its overall power resides in this capability. The method of defining products and ratios may seem very weak at first glance, but recall that once a new term has been defined, the system does not distinguish it from observable terms. Thus, the program can define products of ratios, ratios of products of products, and so forth. Also, upon discovering a linear relation at one level of description, the system treats the slope and intercept as new dependent terms at the next level. This means that slopes and intercepts can themselves be incorporated in complex relations, as we saw in the general version of the ideal gas law above. The ability to introduce intrinsic properties provides power of an entirely different type, letting BACON effectively transform nominal variables into numeric ones, which can then be incorporated into numeric laws.

However, the system is less robust in representing and discovering the scope on laws. We have seen that BACON places conditions – in the form of the values of unvaried terms – on both its laws and its intrinsic values, and that it cautiously drops these conditions if the data merit such action. But one can imagine other alternatives that BACON ignores. For instance, Black's law holds across a broad range of temperatures, but not across the phase boundaries at which substances change from liquid to solid. Similarly, the ideal gas law is an excellent approximation for normal temperatures, but it breaks down at high levels. Ideally, an empirical discovery system should be able to detect and represent such constraints on the laws it formulates.

BACON's ability to generate experiments is also quite limited. The system is presented with independent terms and their suggested values, and from this it algorithmically produces a combinatorial design. There is no sense in which the system gathers data adaptively in response to the observations it makes. Such intelligent experimentation generation is an important component of scientific discovery, and a robust empirical discovery system should have this capacity. In the following section, we examine another system that responds to the issues of scope and experiment generation.

## The FAHRENHEIT System

We have seen that BACON constituted a significant step beyond Gerwin's early discovery work, but that it still had a number of limitations. The most pressing of these revolved around identifying the scope of the discovered laws and generating experiments in an intelligent manner. In this section, we describe Zytkow's FAHRENHEIT (Zytkow, 1987; Koehn & Zytkow, 1988), a successor to BACON that responds to these issues.

### Representing Laws and Their Scope

The FAHRENHEIT system borrows heavily from the earlier work by Langley, Simon, and Bradshaw, including a BACON-like routine as one of its basic components. This component is similar enough to BACON.4 that we will ignore the differences and focus instead on its interaction with the remainder of the system. In other words, Zytkow's work does not

question the basic validity of the earlier system; rather, it argues that BACON told only part of the story. The form of FAHRENHEIT's input is identical to that given to BACON: a set of independent and dependent attributes that take on numeric or symbolic values. Zytkow's program interacts with a separate simulated environment that eases the running of experiments, but this difference is not theoretically significant.

The system's output is also very similar: a set of numeric laws that summarize the data, stated through a set of theoretical terms defined using observables. The existing version does not incorporate intrinsic properties, but these could be easily added. The main difference from BACON lies in the form of the numeric laws. Rather than stating the scope of a law as a simplistic set of independent values, FAHRENHEIT specifies these limits as another set of numeric laws.[3] It accomplishes this feat through a familiar ploy – defining new theoretical terms.

Let us consider a simple form of Black's specific heat law, in which we combine the substances water and mercury and in which we hold their masses constant at 0.1 kg and 5.0 kg, respectively. The simplified law can be stated as: $T_f = jT_M + kT_W$, where $T_M$ and $T_W$ are the initial temperatures for mercury and water, and $T_f$ is the final temperature of both. The terms $j$ and $k$ are constants that hold for this particular pair of substances and the given masses. In fact, this relationship holds only for limited values of the temperature $T_M$ and $T_W$, and it is with representing this limitation that we are concerned.

Like its laws, FAHRENHEIT represents limits on laws at varying levels of description. For instance, suppose the system has formulated the first level law $T_f = aT_M + b$, where $a$ and $b$ are constants for a given temperature $T_W$. Along with these parameters, FAHRENHEIT also defines two *limit* terms, one representing the maximum value of $T_M$ for which the law holds and another for the minimum value. We will call these terms $T_{Mmax}$ and $T_{Mmin}$, respectively.

These limit terms may have different values for different settings of $T_W$, and these values are carried to the second level of description along with $a$ and $b$. At this level, the limit terms themselves may enter into relations with the independent variable. In this case, simple laws exist for both boundary terms: $T_{Mmax} = -0.6T_W + 160$ and $T_{Mmin} = -0.6T_W$. Of course, the system also states laws involving the slope and intercept parameters from the first level; in this case, $a$ is constant and $b = dT_M$.

In addition, FAHRENHEIT also specifies limits on all four of these higher-level laws, defining versions of $T_{Wmax}$ and $T_{Wmin}$, the maximum and minimum temperatures for which each law is valid. This means that the system not only has the ability to place limits on its basic laws; it can also state the boundary conditions under which its boundary laws hold. This is another instance of the recursive theme underlying the class of discovery systems we have

[3] Falkenhainer and Michalski (1986) also address the issue of limiting the scope of laws in their ABACUS system. However, they represent boundaries either as symbolic conditions or as simple maxima and minima on the values of numeric terms.

been considering. Figure 1 summarizes the boundary conditions found in the Black's law example we have just considered.

## Determining the Scope of Laws

Now that we have examined FAHRENHEIT's representation of empirical laws, let us turn to the method by which it discovers them. The system's basic organization is very similar to that used in BACON, and it begins in exactly the same manner – by varying the values of one independent term and examining the effect on the dependent variables. Returning to the Black's law example, suppose FAHRENHEIT varies $T_M$ and observes the resulting values of $T_f$. Using the same heuristics as BACON.4, the program notes a linear relationship between the two terms and formulates the law $T_f = aT_M + b$, where the slope $a = 0.624$ and the intercept $b = 11.28$.

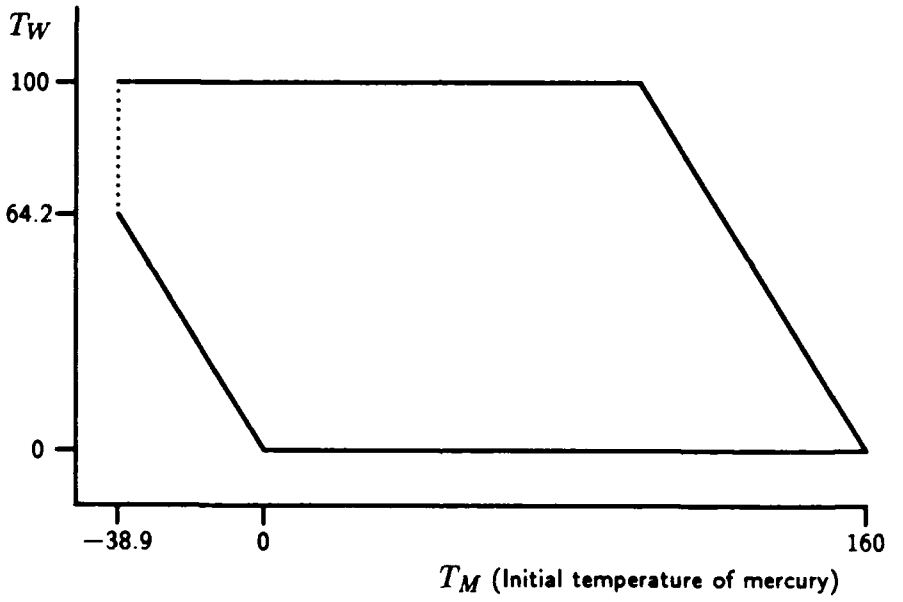



*Figure 1.* The scope of Black's law for 0.1 kg water and 5.0 kg mercury.

At this point, BACON would assume that the only conditions on the new law are the values of the independent terms that have not yet been varied; i.e., that the substances are mercury and water, that $T_W = 30°$, that $M_M = 5kg$, and that $M_W = 0.1kg$. It would proceed to vary these terms in order to determine their effect on the parameters $a$ and $b$. FAHRENHEIT does not make this assumption, realizing that the law relating $T_M$ and $T_f$ may hold for only *some* values of $T_W$. To check this possibility, the system selectively gathers additional data, varying the value of $T_M$ in an attempt to determine upper and lower boundaries on the law.

FAHRENHEIT first increments the independent term by the same user-specified amount used in its earlier data-gathering steps. If the law still holds, it increments by double this amount and checks again. This doubling continues until the system arrives at some value of the variable for which the law is violated, or until it reaches values beyond the range of the measuring instrument. In the latter case, the program assumes the law has no upper limit; in the former case, it attempts to find the exact point at which the law ceases to hold. For this the system uses a successive approximation method, halving the distance between the highest known value that obeys the law and the lowest known value that violates the law. This process continues until it has determined the upper limit within the desired (user-specified) degree of precision. FAHRENHEIT employs the same method to determine the lower limit on the law.[4]

Returning to our example, after discovering the law $T_f = aT_M + b$ when $T_W = 30°$, the system would determine the upper and lower bounds on this law. For this situation, the law holds only between $T_M = 142°$ and $T_M = -18°$; for values outside this range, the linear relation cannot be used to predict the dependent term. Other limits hold for other values of $T_W$, and this leads us to the next stage in FAHRENHEIT's discovery process.

## Discovering Complex Laws and Limits

Recall that once BACON has induced a law relating one independent term to a dependent term (say $T_f$ and $T_M$), it recurses to a higher level. The program varies another independent term (say $T_W$) and, for each value of that term, repeats the experimentation that led to the original law. In each case, the system finds the same form of the law, but the parameters (say $a$ and $b$) in that law may take on different values. These become dependent values at the next higher level of description and are associated with the independent values under which they occurred. Once it has collected enough higher-level data, BACON applies its heuristics to induce a higher-level law (say $a = c$ and $b = dT_W$).

BACON's successor follows the same basic strategy, but as we have seen, it defines two additional theoretical terms for each law discovered at the lower level. The system treats these terms as dependent variables at the next higher level and attempts to relate their values to those of the varied independent term.[5] In our Black's law example, the limit terms are $T_{Mmax}$ and $T_{Mmin}$, whereas the second independent term (to which they must be related) is $T_W$. In this case, FAHRENHEIT discovers the two linear relations described above, one

[4] FAHRENHEIT considers only independent terms in its search for boundary conditions. One can imagine cases in which dependent variables would also be useful, though the resulting laws could not be used for making predictions. One can also envision domains in which the boundaries are not clear-cut; phase boundaries are a good example. To the extent these can be handled as 'noisy boundaries', the system can discover approximate constraints. Extending FAHRENHEIT in both directions is a task for future research.

[5] This means that the number of dependent terms increases by a factor of three, at minimum, for each level ascended. Thus, higher levels of abstraction require that ever more discoveries be made.

between $T_{Mmax}$ and $T_W$ ($T_{Mmax} = -0.6T_W + 160$) and the other between $T_{Mmin}$ and $T_W$ ($T_{Mmin} = -0.6T_W$). They state that, as the temperature of mercury increases, there is a decrease in both the maximum and minimum temperatures of water for which the law holds. These relations are shown as slanted lines in Figure 1. Although both have slopes of $-0.6$, the two lines are independent of each other and have different intercepts.

FAHRENHEIT's next step follows from its inherently recursive nature - it attempts to establish limits on these limit laws. It uses the same scheme it employed at the lower level, exploring values of the independent term (this time $T_W$) until it finds the upper and lower limits on each law. In Figure 1, the upper limit on the maximum law is 100, the upper limit on the minimum law is 64.2, and the lower limit on both laws is zero. The limits for the two laws need not be the same, though the two lower limits are equal (this results from the phase change of water into ice).

However, recall that FAHRENHEIT has also discovered another law at the current level; this is $b = dT_W$, which relates the intercept of the lower-level law to the temperature of water. Naturally, the program also searches for the limits on this law in terms of $T_W$. The lower limit for this law (zero) corresponds to the lower limit for both the maximum and minimum laws, and the upper limit (100) corresponds to the upper limit for the maximum law. However, the latter differs from upper limit (64.2) for the minimum law, indicating a range of the basic law ($a = cT_W$) for which the lower limit is unknown. We have marked this range with a dotted line in the figure. The current version of FAHRENHEIT leaves this range unspecified, but future versions should attempt to determine its functional form as well.

In summary, the new system employs the same recursive structure as BACON, which lets it discover the same higher-level laws (relating multiple variables) as did the earlier system. However, FAHRENHEIT's inclusion of theoretical terms for the scope of a law also lets it discover:

(1) upper and lower limits on the higher-level laws;

(2) laws that express upper and lower limits as functions of other terms; and

(3) limits on these limit-based laws themselves.

The example we have considered is relatively simple in that it involved only two independent terms and thus generated only two levels of description. But FAHRENHEIT's discovery strategy applies equally well to more complex situations involving many variables and levels, and the system will recursively apply its heuristics until it can discover no further regularities.

In the introduction, we reviewed Falkenhainer and Michalski's (1986) ABACUS, a system that determines the scope of laws using a different method. One can view their system as searching for the regions that are composed of hyperectangles in an N-dimensional space, using the Aq algorithm. In contrast, FAHRENHEIT searches for regions bounded by the class of laws that BACON can discover. As a result, the latter system would seem more appropriate for domains with complex boundaries that involve relations between two or more variables.

## Additional Capabilities of FAHRENHEIT

In addition to determining the scope of laws, FAHRENHEIT also includes a number of other abilities beyond those found in BACON. One of these involves irrelevant independent variables. In experiments involving such terms, Langley, Simon, and Bradshaw's system would note a constancy for all dependent variables and state simple laws to this effect. In this sense, BACON could handle irrelevant terms. However, upon coming to a new experimental context involving the same terms, the program would go through the same process of varying the independent term, observing dependent values, noting their constancies, and stating trivial laws. FAHRENHEIT avoids this extra effort and unneeded data-gathering by marking such independent attributes as irrelevant and bypassing them in later experiments. This can lead to substantial savings, especially if the irrelevant terms are ones that would have been varied earlier in the discovery process and thus would have been included at lower levels of description.[6]

Another of BACON's limitations involved the order in which irrelevant terms were varied. Although in many cases the system was insensitive to the order, this did not hold for some of the more complex laws. Let us return to Black's law for an example. In its full form, this law relates the final temperature $T_f$ not only to the initial temperatures $T_1$ and $T_2$ of the combined substances, but also to the masses $M_1$ and $M_2$ of those substances. In the reported runs on Black's law (Langley, Bradshaw, & Simon, 1983; Langley, Simon, Bradshaw, & Zytkow, 1987), the temperatures were always varied first, but let us examine the result when the masses are used instead.

Suppose we place two containers of water into contact, with $T_1 = 20°$, $T_2 = 40°$, and $M_1 = 1$. Upon varying the mass of the second container $M_2$ and observing the resulting values of $T_f$, we obtain data that obey the law $T_f = 20(1+2M_2)/(1+M_2)$. However, BACON's heuristics are not powerful enough to discover this law. When we tell the system to vary the independent terms in this order, it will fail to recognize any regularity in the resulting data. One response would be to replace BACON's law-finding rules with more powerful curve-fitting methods, but this is sidestepping the real issue. Any law-finding method will have some limits, and these limits will eventually emerge when encountering the right order of variation.

FAHRENHEIT responds to this possibility by considering different orders of varying the independent terms. The system operates in normal Baconian mode until it encounters some term that appears relevant, but for which it cannot find any regular law. In such cases, the program sidesteps the variable and places it at the end of the queue to ensure that it will be reconsidered later. It then varies the next independent term in the queue and attempts to incorporate this variable into some law. If this also fails, FAHRENHEIT considers the next

[6] Langley (1981) describes an earlier version of BACON that identified irrelevant terms in a similar manner and modified its experiments in response. These abilities were dropped in later versions to devote attention to other issues.

term, and so forth. If it cannot find laws for any of the remaining independent terms, the system halts with only a partial law.

Although this strategy is more robust than BACON's method and can handle the Black's law example given above, it does not consider all possible orders and thus is not guaranteed to find the maximal laws. FAHRENHEIT's authors have experimented with a variant on the above algorithm that, upon failing to find laws for any of the remaining terms, backtracks to consider different orders of variables that have already been successfully related. This scheme is more complete, but it is also more expensive. In the worst case, the simpler method has a computational complexity of $N(N+1)/2$, where $N$ is the number of independent terms. In contrast, the backtracking method has a worst case complexity of $N!$, though we doubt this would occur very often.

### Evaluating FAHRENHEIT

We have seen that FAHRENHEIT introduced a number of improvements over BACON. The system's ability to consider alternative orders of varying independent variables lets it discover laws under conditions in which BACON would have failed. The program also handles irrelevant terms in a more sensible way than its precursor, leading to savings in both time and in the amount of data required.

Most important, FAHRENHEIT represents the scope of laws in a more robust manner than did BACON, and it incorporates heuristics to discover such limits in scope. This requires a more intelligent data-gathering strategy than was present in the earlier program, involving the selectively generated experiments that depend on the results of earlier experiments. Moreover, FAHRENHEIT does not halt upon finding the upper and lower limits on a law; it defines theoretical terms for these limits and carries them to the next level of description, along with the parameters for its basic laws. Using its recursive structure, the system then searches for laws relating these limit terms to new independent variables, and searches for limits on these laws in turn.

The resulting limits and limit-related laws establish a clear context in which FAHRENHEIT believes its basic laws to hold. But this context is still based largely on 'number games,' and it tells us little about the qualitative structure of situations in which the laws are valid. In the following section, we will see another approach to representing and discovering the context on empirical laws that responds to this issue.

## The IDS System

Langley and Nordhausen (1986) have described IDS, an integrated discovery system that formulates both qualitative laws and discovers numeric relationships. Although this program is superficially responding to the same task as the other systems we have examined, it differs significantly in both its representation of laws and in its discovery process. This research effort is still in its early stages, but in this section we report the progress to date. As before, we will begin by considering representational issues.

## The Need for Qualitative Descriptions

Like FAHRENHEIT, the IDS system interacts with a simulated world in which it can run experiments and gather data. But this environment differs from Zytkow's simulation in two important ways: (1) all attribute-value pairs are associated with specific objects; and (2) the values of these attributes change over time. Also, the program interacts with its world through an explicit set of sensors (that measure the value of a particular attribute for a particular object) and a set of effectors (that carry out primitive actions, such as moving or heating objects). Thus, IDS has available to it a more realistic environment than earlier systems, and this is reflected in its representation of laws.

This representation is best explained through an example, and since we have used Black's law earlier in the paper, let us consider it again. The previous versions of this law, as represented by BACON and FAHRENHEIT, related the masses ($M_1$ and $M_2$), specific heats ($c_1$ and $c_2$), and initial temperatures ($T_1$ and $T_2$) of two substances to their final temperature ($T_f$) after they had been in contact for some time. However, Black's law actually involves much more than this single equation. Let us walk through what actually transpires in such an experiment.

We begin with two substances, having known masses and stable temperatures, which are then placed in contact. If we measure the temperatures over time, we will observe that the higher one gradually decreases and the lower one gradually increases. This process continues until the two temperatures become equal, at which point both remain constant. Note that much of the interesting detail in this example is lost in the BACON/FAHRENHEIT representation. Some might be regained by including separate final temperatures for each object, but there would still be no sense of two quantities gradually moving towards equilibrium.

## Representing Change with Qualitative Schemas

IDS is able to represent such knowledge by using *qualitative schemas* that summarize changes over time in the values of one or more objects. A schema consists of a finite state diagram in which successive states represent succeeding intervals of time. For instance, the schema for Black's law contains three such states: the first describes temperatures before contact; the second describes temperatures after contact but before equilibrium is reached; and the last describes temperatures after the physical system achieves equilibrium. Figure 2 presents a graphical description of this three-state schema.

Each state in the schema has an associated description of the observed attributes. These descriptions state whether a given attribute's values are increasing, decreasing, or constant during that state. In fact, these 'qualitative derivatives' define the boundaries of each state. In matching the schema against a new instance of Black's law, IDS knows when the physical system has moved into the next state by noting when the signs of the various derivatives change. For instance, the second state in the figure applies only to those time steps in which the first object's temperature is increasing (the qualitative derivative is positive) and the second object's temperature is decreasing (the derivative is negative).

This knowledge representation is very similar to that suggested by Forbus (1984) in his qualitative process (QP) theory, and we have been strongly influenced by this work. Given a set of physical processes and some initial description of the environment, QP theory describes how one can generate an *envisionment* of the states the physical system will enter as those processes operate. The qualitative schemas of IDS are nearly identical to Forbus' envisionments. We have used a different term because IDS induces its schemas directly from observations, whereas in qualitative process theory, envisionments are deduced from process descriptions. We do not have the space to describe the generality of this approach to representing physical systems, but it can be used to provide qualitative descriptions for a substantial range of phenomena from both physics and chemistry. For this reason, we believe it provides an excellent basis for machine discovery.

## Inducing Qualitative Schemas

In its initial knowledge state, IDS contains a simple qualitative schema for each of its effectors. For instance, the initial schema for the heat effector includes two states: one in which a heater near an object is turned off (and in which the object's temperature is constant); and another in which the heater is turned on (and in which the object's temperature is increasing). The initial schema for placing objects in contact is even simpler; IDS expects that the only effect of placing one object adjacent to another is to change its location.

However, experiments with objects having different temperatures lead to violated expectations, and these in turn cause IDS to modify the qualitative structure of this second schema. We do not have the space to detail the processes used in acquiring qualitative schemas, but we can list the three basic methods:

- If IDS encounters entirely new behavior, it creates a new state and adds this to the current schema.
- If the system recognizes itself in a known state that was not predicted, it adds a connection between this state and the previous one.
- Upon finding evidence that a state's description is overly general, IDS makes that description more specific.

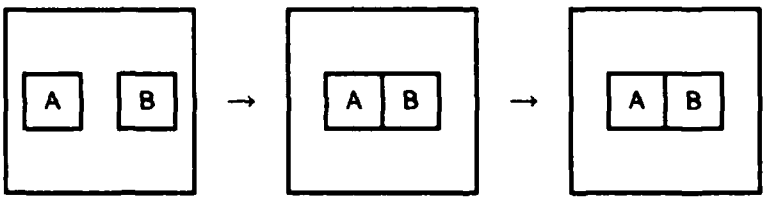
Taken together, these methods let the program incrementally improve its qualitative schemas as it gains more experience with its environment. They lead from the initial 'place-in-contact' schema to the schema shown in Figure 2. This latter schema accurately describes the qualitative structure underlying Black's law.

## Embedding Numeric Laws in Qualitative Schemas

We have focused on representing qualitative knowledge in IDS, but the system can also state numeric laws. The form of these laws is intimately related to the structure of the schemas, which are both object-oriented and time-oriented. As a result, numeric terms are specified using two subscripts, one for the object involved and another for the state. In the

Black's law schema, the temperature of the first object (A) in the second state would be $T_{A,2}$, whereas the temperature of the second object (B) in the third (final) state would be $T_{B,3}$. Thus, IDS would state the numeric aspects of Black's law as $T_{A,3} = (c_A M_A T_{A,2} + c_B M_B T_{B,2})/(c_A M_A + c_B M_B)$ and $T_{B,3} = T_{A,3}$.

Qualitative schemas serve two main purposes with respect to numeric laws. First, they provide a context within which the law has meaning. Clearly, if one places two objects into contact and they do not obey the qualitative structure of the schema in Figure 2, then one would not expect their quantitative relations to obey Black's law. For example, some substances might be so well insulated that their temperature loss is negligible. This approach also lets one qualitatively handle phase shifts, which FAHRENHEIT modeled in a purely quantitative fashion.




| State 1 | State 2 | State 3 |
|---|---|---|
| $dist(A,B) > 0$ | $dist(A,B) = 0$ | $dist(A,B) = 0$ |
| $T(A) > T(B)$ | $T(A) > T(B)$ | $T(A) = T(B)$ |
| $\Delta T(A) = 0$ | $\Delta T(A) < 0$ | $\Delta T(A) = 0$ |
| $\Delta T(B) = 0$ | $\Delta T(B) > 0$ | $\Delta T(B) = 0$ |

*Figure 2.* A qualitative schema for Black's law.

Second, qualitative schemas constrain the search for numeric laws and provide the basis for designing and running systematic experiments. The simulated environment provides IDS with effectors that let it generate objects of desired mass and let it alter the temperature of those objects. Given this ability, the system can 'run' the Black's law schema under different initial conditions and observe the results. Each such 'run' provides the system with a different set of data, but these data are much more structured than those available to BACON or FAHRENHEIT.

IDS applies data-driven heuristics to these numeric data in hopes of finding constant terms and simple linear relations. In the Black's law example, it follows a path much like that taken by BACON, though the terms involved are slightly different. One important difference is that *all* terms that are constant throughout the schema can be viewed as intrinsic properties of the objects used in the experiment. Moreover, when IDS runs the same experiment using different objects, different substances, or different classes of substances, it may discover the same values for these terms. In such cases, it raises the retrieval conditions for the intrinsic value to the appropriate level of generality. In this framework, intrinsic values are associated directly with objects or classes of objects, not with nominal values themselves.

In addition, new types of intrinsic properties arise within the IDS scheme that do not appear within earlier approaches. The system may note that the schema shifts from one state to its successor whenever the value of a particular term reaches a certain value. The introduction of such *limit conditions* lets one represent and discover concepts like melting points and boiling points, which signal changes of qualitative state. IDS may also note that the *duration* of a state is constant or that it is related to some other term. This leads to concepts such as latent heat and the heat of vaporization. All of these concepts are types of intrinsic property, with different values associated with different substances. Thus, the use of qualitative schemas provides representational support for intrinsic terms that could not be handled in earlier frameworks.

### Evaluating IDS

Our research on IDS is still in its early stages, though we have a running system that we have tested on a variety of heat-related laws. But the representational power of the system seems considerably greater than that of BACON, and it provides more context for its numeric laws than does FAHRENHEIT. But that does not mean that one system is superior to the other. The current version of IDS does not include the methods for determining scope that Zytkow's system introduced, and these should definitely be considered for future versions. Nor does the system consider different orders of independent variables, though the use of qualitative schemas significantly simplifies the search for numeric relations.

We would argue that IDS's greatest significance lies in its attempt to integrate the discovery of qualitative and quantitative laws. Earlier work has focused on one or the other, but has not considered their combination.[7] One ultimate goal of research in machine discovery is the construction of an integrated discovery system that covers many aspects of scientific reasoning, and we believe IDS is an important step in that direction.

## Perspectives on Machine Discovery

Having sampled the evolution of research on data-driven approaches to machine discovery, let us turn to the implications of this research. In any area of study, we find two distinct but complementary views, one concerned with description and the other concerned with prescription. In the study of scientific reasoning, historians of science take the first perspective, while philosophers of science take the second. We close the paper by considering the relevance of machine discovery to these areas, devoting more space to the normative side.

### Machine Discovery and the History of Science

The history of science studies the actual path followed by scientists over the years, attempting to understand the steps taken toward a particular scientific advance, along with

[7] Falkenhainer and Michalski's (1986) ABACUS identifies symbolic conditions on numeric laws, but their system does not discover qualitative laws of the type formulated by GLAUBER (Jones, 1986; Langley et al., 1986, 1987).

the reasons for those steps. Traditionally, historians of science have been content with verbal descriptions of scientific behavior, but the advent of machine discovery systems suggests an alternative: one can view AI discovery systems as computational models of the historical discovery process.

Whether or not they provide *adequate* models is partly a matter of one's goals. In their early work on computational models of human problem solving, Newell and Simon (1972) argued for the usefulness of *sufficient* models of behavior. Such models do not account for the details of human behavior on a task, but they do have roughly the same capabilities. Once such models have been developed, they may be replaced with more careful simulations. We would argue that BACON and its successors provide such sufficient models of empirical discovery. On close inspection, we find that the detailed behavior of early scientists like Ohm, Coulomb, Black, and others diverges from that followed by the programs.[8] Nevertheless, these systems have shown themselves capable of discovering the same laws as the scientists, and this provides an excellent starting point for more detailed computational models.

Within both frameworks, one can take two approaches to testing the adequacy of discovery models. The most common technique involves arguing for the model's generality by showing it can discover a wide range of laws with a variety of forms. This is the approach taken with BACON, and Falkenhainer and Michalski (1986) have evaluated their ABACUS system along similar lines. The other approach involves running the model on an extended example that consists of a lengthy sequence of discoveries. This is the approach taken by Lenat (1977) with his AM system, and Nordhausen and Langley have used a similar strategy in testing IDS. To the extent that the system's steps follow the same path that was taken historically, one can argue that it constitutes a plausible model of historical discovery.

Although none of the systems that we have described give an acceptable detailed account of historical discoveries, we believe they provide an excellent framework for future work in this direction. Whether such efforts should have high priority is an open question. Clearly, much more remains to be done in developing sufficient models of the discovery process, but this does not exclude the development of detailed models by other researchers. In many ways the latter is more difficult, since it requires intimate familiarity with historical developments. However, this road must ultimately be taken if we hope to formulate complete descriptive theories of scientific discovery.

## Validation and Discovery

Historically, the nature of induction and discovery have played an important role in the philosophy of science. Early contributors such as Sir Francis Bacon (1620) and John Stuart Mill (1843) proposed 'logics of induction' as methods for uncovering scientific laws. However, with the advent of the 20th Century this interest passed, and most philosophers turned their attention to the *validation* of scientific laws and theories. Indeed, some researchers

[8] For an example of a more careful model of historical discovery, see Zytkow and Simon's (1986) description of their STAHL system.

even argued that a 'logic of discovery' was impossible (Popper, 1961). Recently some have regained interest in the topic of discovery (Nickles, 1978), but the mainstream has retained its skepticism about the normative study of discovery.

Let us consider more closely what is meant by a normative or prescriptive theory of discovery.[9] Obviously, it suggests a set of methods that one *should* follow in formulating scientific laws. For those with a logical bent, this may translate as 'a deductively valid set of methods,' and we agree that no such methods are possible; inductive inference is clearly not deductively valid. However, this definition seems overly constraining. We cannot expect inductive techniques to give us *correct* laws, but we might legitimately require them to provide *useful* laws. Let us see what this might mean; the philosophy of science itself provides several responses, as we discuss in the following sections.

## Eliminability of Theoretical Terms

The nature of theoretical terms has occupied a central role in recent philosophy of science. One important result involves the notion of *eliminability.* A theoretical term is said to be *eliminable* if one can replace all of its occurrences in a theory with directly observable terms. A theory containing only eliminable terms can be tested in a straightforward manner, by simply replacing these terms with observables and comparing its predictions against the data. In contrast, theories that contain non-eliminable terms cannot always be tested, giving them questionable status.

Given this view, one might want a discovery method that introduces noneliminable terms into its laws and theories only as a last resort. We have seen the role played by theoretical terms in BACON, FAHRENHEIT, and IDS. Some of these terms, such as the product $PV$ in the ideal gas law, are defined directly using observable variables. Others, such as the intrinsic property of specific heat in Black's law and FAHRENHEIT's limit terms, are defined in a more roundabout manner. However, all such terms can be eliminated from the law and replaced with direct observables. In this sense, the systems we have examined employ normative discovery methods.

## Laws and Definitions

Another issue involves Glymour's (1980) criterion of *bootstrap confirmation.* Many philosophers have made a strong distinction between *definitions* and *laws*, with only the latter having empirical content. Glymour argues against this dichotomy, claiming that it is the combination of laws and definitions that have empirical content, and that these combinations can be tested against the same type of data used in generating them.

This is exactly the situation that occurs when BACON postulates an intrinsic property. In the Black's law example, we saw that when the system initially proposed the property of specific heat, the assigned numeric values were tautologically defined. At this point, the law

[9] For a detailed analysis of normative and descriptive systems of discovery and logic of discovery, see Zytkow and Simon (1988).

involving the new term had no empirical content; it was guaranteed to hold. However, as new data were gathered, there was no (deductive) reason to expect it to apply in these cases. Had it failed to describe the new observations, the law would have been disconfirmed. In this case, it successfully covered the data and so was retained, but that is not the issue. Rather, the point is that what begins as a definition with no empirical content can be tested as additional data become available. This is the essence of Glymour's bootstrapping criterion, and to the extent that BACON's methods incorporate this criterion, it can be viewed as a normative theory of discovery.

### Optimal Laws and Heuristic Search

The notions of eliminability and bootstrapping place constraints on laws and theories, but they do not specify which laws are optimal for a given set of data. Some proposals have been made for such criteria. For instance, Popper (1961) has suggested that more falsifiable theories should be preferred to less easily rejected ones. Other suggestions have invoked the notions of simplicity and fertility. We feel that stating such criteria is a useful task for both machine discovery and philosophers of science, but even if researchers could agree on such criteria, we would not insist that a normative theory be able to achieve such optimal laws.

One of the central insights of AI is that intelligence involves search through combinatorial spaces, and that one can seldom afford to search these spaces exhaustively. Instead, one must employ heuristic methods that cannot guarantee optimal solutions, but which are reasonably efficient. As Simon (1956) has argued, one must often be content with solutions that *satisfice* for a given problem. This means that realistic discovery methods cannot guarantee the generation of optimal laws and theories, even if the criteria for such optimality are clearly defined. Instead, a normative theory of discovery should generate laws that approximate these criteria.

Progress can occur in prescriptive fields just as it can in descriptive ones. The fact that BACON and its successors constitute normative theories of discovery does not mean they are the best such theories. For example, Kokar (1986) has argued that his COPER method is superior to the BACON approach along a number of dimensions, and Falkenhainer and Michalski (1986) have made similar claims about their ABACUS system. Future work in machine discovery and the philosophy of science may produce improved logics of discovery. Such improvements should be measured by the degree to which a normative theory produces laws that account for existing data and correctly predict new observations.

## Conclusion

In this paper we addressed the task of empirical discovery, focusing on four AI systems that share a common approach to inducing numeric laws. This approach relies on data-driven heuristics, the definition of theoretical terms, and the recursive application of a few basic methods. We saw that Langley, Bradshaw, and Simon's BACON system introduced some important advances over Gerwin's earlier work, including the ability to handle multi-

ple independent terms and to postulate intrinsic properties. We also found that Zytkow's FAHRENHEIT system incorporated some significant methods not present in BACON, such as the ability to determine the scope of laws and the ability to consider different orders of varying independent terms. Finally, we saw that Nordhausen and Langley's IDS is able to embed numeric laws within a qualitative description, providing significantly more context for these laws than earlier systems.

In the last section, we examined these systems from two perspectives: as models of historical discovery and as normative theories of discovery. We found that these particular systems do not fare well as detailed historical models, though they provide good starting points for the development of improved models. We also found that the existing systems can be viewed as normative models of how empirical discovery *should* proceed, though we also argued that future systems will provide improved norms for inductive behavior.

As we stated at the outset, empirical discovery is only one part of the complex phenomenon that we call science. But we feel that it is an important part, and that the systems we have described constitute a significant step towards understanding the nature of scientific discovery. Recent work has started to address other aspects of the scientific process, including theory formation (Falkenhainer, 1987; Shrager, 1987), theory revision (Rose & Langley, 1986), and experimentation (Kulkarni & Simon, 1988). We expect future work to extend these promising efforts on isolated aspects of science. However, we also expect researchers to develop *integrated* models that combine many aspects of the discovery process, and we would be surprised if they did not incorporate at least some ideas from the work that we have examined here.

## References

Bacon, F. (1960). *The New Organon and related writings* (F. H. Anderson, ed.). New York: Liberal Arts Press.

Bradshaw, G. L., Langley, P., & Simon, H. A. (1980). BACON.4: The discovery of intrinsic properties. *Proceedings of the Third Biennial Conference of the Canadian Society for Computational Studies of Intelligence* (pp. 19–25). Victoria, B.C., Canada.

Dietterich, T. G., & Michalski, R. S. (1983). A comparative review of selected methods for learning from examples. In R. S. Michalski, J. G. Carbonell, & T. M. Mitchell (Eds.), *Machine learning: An artificial intelligence approach.* Los Altos, CA: Morgan Kaufmann.

Falkenhainer, B. C. (1987). Scientific theory formation through analogical inference. *Proceedings of the Fourth International Workshop on Machine Learning* (pp. 218–229). Irvine, CA: Morgan Kaufmann.

Falkenhainer, B. C., & Michalski, R. S. (1986). Integrating quantitative and qualitative discovery: The ABACUS system. *Machine Learning, 1*, 367–401.

Fisher, D. (1987). Knowledge acquisition via incremental conceptual clustering. *Machine Learning, 2*, 139–172.

Forbus, K. D. (1984). Qualitative process theory. In D. G. Bobrow (Ed.), *Qualitative reasoning about physical systems* (pp. 85–168). Cambridge, MA: MIT Press.

Gerwin, D. G. (1974). Information processing, data inferences, and scientific generalization. *Behavioral Science, 19*, 314–325.

Glymour, C. (1980). *Theory and evidence.* Princeton, NJ: Princeton University Press.

Jones, R. (1986). Generating predictions to aid the scientific discovery process. *Proceedings of the Fifth National Conference on Artificial Intelligence* (pp. 513–517). Philadelphia, PA: Morgan Kaufmann.

Koehn, B. W., & Zytkow, J. M. (1988) Experimenting and theorizing in theory formation. *Proceedings of the ACM Sigart International Symposium on Methodologies for Intelligent Systems* (pp. 296–307). Knoxville, TN.

Kokar, M. M. (1986). Determining arguments of invariant functional descriptions. *Machine Learning, 1*, 403–422.

Kulkarni, D., & Simon, H. A. (1988). The process of scientific discovery: The strategy of experimentation. *Cognitive Science, 12*, 139–175.

Langley, P. (1978). BACON.1: A general discovery system. *Proceedings of the Second Biennial Conference of the Canadian Society for Computational Studies of Intelligence* (pp. 173–180). Toronto, Ontario, Canada.

Langley, P. (1981). Data-driven discovery of physical laws. *Cognitive Science, 5*, 31–54.

Langley, P., Bradshaw, G. L., & Simon, H. A. (1983). Rediscovering chemistry with the BACON system. In R. S. Michalski, J. G. Carbonell, & T. M. Mitchell (Eds.), *Machine learning: An artificial intelligence approach.* Los Altos, CA: Morgan Kaufmann.

Langley, P., & Jones, R. (1988). A computational model of scientific insight. In R. Sternberg (Ed.), *The nature of creativity.* Cambridge: Cambridge University Press.

Langley, P., & Michalski, R. S. (1986). Machine learning and discovery. *Machine Learning, 1*, 363–366.

Langley, P., & Nordhausen, B. (1986). A framework for empirical discovery. *Proceedings of the International Meeting on Advances in Learning.* Les Arc, France.

Langley, P., Simon, H. A., Bradshaw, G. L., & Zytkow, J. M. (1987). *Scientific discovery: Computational explorations of the creative processes.* Cambridge, MA: MIT Press.

Langley, P., Zytkow, J. M., Simon, H. A., & Bradshaw, G. L. (1986). The search for regularity: Four aspects of scientific discovery. In R. S. Michalski, J. G. Carbonell, & T. M. Mitchell (Eds.), *Machine learning: An artificial intelligence approach* (Vol. 2). Los Altos, CA: Morgan Kaufmann.

Lenat, D. B. (1977). Automated theory formation in mathematics. *Proceedings of the Fifth International Joint Conference on Artificial Intelligence* (pp. 833–842). Cambridge, MA: Morgan Kaufmann.

Michalski, R. S., & Larson, J. B. (1978). *Selection of most representative training examples and incremental generation of* VL1 *hypotheses: The underlying methodology and description of programs* ESEL *and* AQ11 (Technical Report 867). Urbana: University of Illinois, Department of Computer Science.

Michalski, R. S., & Stepp, R. (1983). Learning from observation: Conceptual clustering. In R. S. Michalski, J. G. Carbonell, & T. M. Mitchell (Eds.), *Machine learning: An artificial intelligence approach.* Los Altos, CA: Morgan Kaufmann.

Mill, J. S. (1974). *Philosophy of scientific method.* New York: Hafner Press.

Mostow, J. D. (1983). Machine transformation of advice into a heuristic search procedure. In R. S. Michalski, J. G. Carbonell, & T. M. Mitchell (Eds.), *Machine learning: An artificial intelligence approach.* Los Altos, CA: Morgan Kaufmann.

Newell, A., & Simon, H. A. (1972). *Human problem solving.* Englewood Cliffs, NJ: Prentice-Hall.

Nickles, T. (ed.). (1978). *Scientific discovery, logic, and rationality.* Dordrecht: Reidel.

Popper, K. (1961). *The logic of scientific discovery.* New York: Science Editions.

Rose, D., & Langley, P. (1986). Chemical discovery as belief revision. *Machine Learning, 1,* 423–451.

Shrager, J. (1987). Theory change via view application in instructionless learning. *Machine Learning, 2,* 247–276.

Simon, H. A. (1956). Rational choice and the structure of the environment. *Psychological Review, 63,* 129–138.

Zytkow, J. M. (1987). Combining many searches in the FAHRENHEIT discovery system. *Proceedings of the Fourth International Workshop on Machine Learning* (pp. 281–287). Irvine, CA: Morgan Kaufmann.

Zytkow, J. M., & Simon, H. A. (1986). A theory of historical discovery: The construction of componential models. *Machine Learning, 1,* 107–136.

Zytkow, J. M., & Simon, H. A. (1988). Normative systems of discovery and logic of search. *Synthese, 74,* 65–90.